Version 1.1 — — — — — — — — Last updated on June 18, 2023

The focus of the proposed project should be **database-centric**. In particular, the database must be **structured and relational**. You are encouraged to choose a project which requires quite a number of entities and complex relationships among them.

Tentative breakdown of marks distribution.

SL	Item	Description	Mark (%)
1	ERD Diagram	Approved by Supervisor and revised if necessary. - Proper representation of entities and relationships - Correctly identifying primary keys	20%
2	Schema and Database Creation	- Proper usage of primary, foreign, and composite keys. - There should be at least one table with composite keys - Default values should be used where necessary - There should be crated_on and last_updated_on fields for appropriate tables. - Indexes should be created wherever appropriate. - Constraints should be imposed wherever appropriate.	15%
3	Representative Data Insertion	1. The DB should have a large number of entries. At least on the scale of thousands. 2. You can write custom scripts to insert those entries. There are also some libraries available to generate and insert dummy data. 3. [*Note] This requirement could be relaxed for some groups based on their project's requirement	0-5%
4	Basic Queries	1. At least 5 INSERT queries. 2. At least 2 user actions that cause insertions spanning 3 or	15%

		more tables. 3. At least 3 UPDATE queries. 4. At least 2 user actions that cause updates spanning 2 or more tables. 5. The user must be able to select parameters for these queries from the interface. 6. At least 3 DELETE queries 7. At least 1 delete query should initiate cascade delete. 8. During deletion, some entries might be required to be "soft deleted". It will depend on the specific project's requirements.	
5	Advanced Queries (Join, Subquery, Set Operations etc)	1. At least 5 such queries 2. At least 2 queries should join 3 or more tables 3. The user must be able to select parameters for these queries from the interface.	15%
6	PL/SQL	1. At least 3 triggers 2. At least 3 stored procedures. If possible, within the project's requirements, Try to find a feature that requires bulk processing and apply stored procedures for that task. For example, a procedure may calculate CGPA for all (e.g., 100000) students. 3. At least 2 functions. For example, a function could be written to hash the user password using ORA_HASH to store and match in the user table. 4. Procedure and function calls should be initiated from the user interface. 5. Procedures and function calls should be tracked through another table for accounting and auditing.	20%
7	Overall Project's	- How well organized is the	10 - 15%

	functionality and responsiveness	project? - How does it perform as a whole? - UI and UX.	
8	Bonus	For exceptional contributions, the supervisor may award bonus points. Example: - A dedicated table to log which procedure or function is called when, by which user, and with which parameter(s). This data will be available from the admin side of the application for accounting and auditing purposes. - Statistics, accounting, and auditing-related queries provided for the admin panel.	
9	Total		100%

Useful data sources for statistics-type projects:

1. https://data.worldbank.org/
2. https://www.who.int/data/gho/
3. https://www.census.gov/data.html’
4. https://www.imf.org/en/Data
5. https://developer.imdb.com/non-commercial-datasets/
6. https://ogirardot.wordpress.com/2013/01/31/sharing-pypimaven-dependency-data/
7. https://gist.github.com/waldoj/5053946
8. https://dblp.org/faq/How+can+I+download+the+whole+dblp+dataset.html

Examples of some previous projects can be found here. But the sky is your limit!